JOURNAL OF COMPUTER AND SYSTEM SCIENCES **53**, 26–32 (1996)
ARTICLE NO. 0047

Learnability: Admissible, Co-Finite, and Hypersimple Languages

Ganesh Baliga*

Department of Computer Science, Rowan College of New Jersey, Glassboro, New Jersey 08028

and

John Case

Department of Computer and Information Sciences, University of Delaware, Newark, Delaware 19716

Received June 9, 1993; revised December 18, 1995

Presented is a surprising characterization of hypersimple sets in algorithmic learning theory. It is used herein to obtain an elegant, tight separation result for learnability criteria. It is argued that such separation results may yield insight for eventual characterizations.

1. INTRODUCTION

In Gold's paradigm [Gol67] for language learning (or invariants of it [OSW86, Cas88]), an algorithmic device is fed *positive* membership information about a (formal) language, and it attempts eventually to conjecture reasonably accurate grammars for that language. This eventual algorithmic learning of reasonably accurate grammars for languages from positive information is difficult, yet Gold [Gol67] cites [McN66] for psycholinguistic evidence that positive information seems to be enough for people. Beginning just below, we will present some examples of what can and cannot be learned from positive information. These examples will help us introduce the results of the present paper.

We suppose for convenience and without loss of generality that every language consists of nothing but non-negative integers. Let $\mathscr{FIN}$ be the class of all finite languages. From [Gol67], for $\mathscr{L} = \mathscr{FIN}$, there is an algorithmic device d such that

> given any language in $\mathscr{L}$, d eventually converges to a single final, perfectly correct grammar generating that language. (1)

Suppose $L \notin \mathscr{FIN}$. By contrast, for $\mathscr{L} = \mathscr{FIN} \cup \{L\}$, there is no algorithmic device d such that (1) [Gol67, OSW86]. We call $\mathscr{FIN}$ *supersaturated* because of this contrast: learnability breaks down if *any* single language not in $\mathscr{FIN}$ is added to it; no language is *admissible* to $\mathscr{FIN}$ without loss of learning power.

Some classes of languages $\mathscr{L}$ are merely *saturated*: adding *some* r.e. $L \notin \mathscr{L}$ to such $\mathscr{L}$ results in a loss of learning power. For saturated classes some r.e. languages are not admissible, but others may be. An example will be presented shortly below and again in Corollary 1.

A language L is co-finite $\stackrel{\text{def}}{\Leftrightarrow}$ it is missing at most finitely many non-negative integers, i.e., $\stackrel{\text{def}}{\Leftrightarrow}$ it is a finite variant of N, the entire set of non-negative integers. Let $\mathscr{COF}$ = the class of all cofinite languages. For $\mathscr{L} = \mathscr{COF}$, there is no algorithmic device d such that (1) [OW82a, OSW86]. Hence, even a class of languages as simple as $\mathscr{COF}$ is difficult to learn from positive information. However, since every element of $\mathscr{COF}$ is a finite variant of N, $\mathscr{COF}$ can be learned if we relax a bit our criterion for successful learning: for $\mathscr{L} = \mathscr{COF}$, there is an algorithmic device d such that

> given any language in $\mathscr{L}$, d eventually converges to a single final grammar which generates a finite variant of that language. (2)

Hence, $\mathscr{COF}$ witnesses that more can be learned with respect to the criterion of success specified by (2) than from that specified by (1) [OW82a, OSW86]. We say, then, that $\mathscr{COF}$ witnesses a *separation* result for learning criteria.

More generally, and in some cases surprisingly, simple subclasses of $\mathscr{COF}$ witness separation results for learning criteria. An example from [CL82, Cas92] appears in (4) at the beginning of Section 3 below. In some cases, separation results, originally obtained by self-referential examples, can be witnessed more simply by natural subclasses of $\mathscr{COF}$. An example such separation, which improves on self referential separation witnesses in [Cas88, Cas92], is presented as Theorem 2 in Section 3 below.

We show (Corollary 1 in Section 3 below) that, with respect to the learning criterion specified by (2), $\mathscr{COF}$ is saturated, but not supersaturated.

* The respective email addresses of the authors are 'baliga@rowan.edu' and 'case@cis.udel.edu'. This paper is an expansion of the conference article [BC93].

0022-0000/96 $18.00

We say that a language L is *co-infinite* $\overset{\text{def}}{\Leftrightarrow}$ it is not co-finite. A language L is called *hypersimple* $\overset{\text{def}}{\Leftrightarrow}$ [L is recursively enumerable and co-infinite $\wedge$ the function enumerating the complement of L in increasing order is not bounded above by any computable function] [Rog67]. Post [Pos44] originally defined hypersimple sets by a different characterization which is presented below in Definition 1 (Section 2.1) and exploited in the proof of our main theorem (Theorem 3 in Section 3 below). This theorem provides a very surprising characterization of hypersimple sets within learnability theory. It says that, with respect to the learning criterion specified by (2), for r.e. L, L is inadmissible to $\mathscr{COF}$ (i.e., L witnesses that $\mathscr{COF}$ is saturated) $\Leftrightarrow L$ is hypersimple. In Corollary 4 below in Section 3 we use $\mathscr{COF} \cup \{L\}$, where L is hypersimple, as a particularly elegant witness to a learning criteria separation result from [OW82a, CL82].

Why are hypersimple sets involved in a separation result in learnability? Actually, hypersimple sets seem, more generally, to capture an essence of many separation and independence results in a wide diversity of contexts besides learning theory. Originally hypersimple sets were used to separate T-complete sets from tt-complete sets [Pos44, Rog67]. Hypersimple sets are crucial in a surprising characterization of recursively axiomatizable theories that have no independent recursive axiomatization [Kre57, PE68]. Furthermore, hypersimple sets have played a role in independence results for both complexity theory [JY81] and control structures [Roy87]. We expect that an important understanding of the role of hypersimple sets in separation and independence in general can be obtained by a careful comparative study of the examples referenced in this paragraph.

In Theorem 5 we present a self-referential witness to a saturation result for a case where $\mathscr{COF}$ would not work. N is the inadmissible set we use to witness this saturation result. Saturated (or supersaturated) classes are just barely inside the boundary of the power of the underlying learning criteria, and the inadmissible sets used to witness the saturation are just on the other side of that boundary. We believe, then, that further study of saturated classes and corresponding inadmissible sets will give us a greater understanding of the underlying learning criteria, and, may, in some cases, lead to insightful characterizations.

Lastly Theorem 6 provides a characterization of the r.e. not recursive sets within learnability theory and yields as a corollary (Corollary 6) another separation result from [OW82a, CL82].

2. PRELIMINARIES

2.1. *Notation*

Any unexplained recursion theoretic notation is from [Rog67]. N denotes the set of natural numbers, $\{0, 1, 2, 3, ...\}$. Unless otherwise specified, e, i, j, k, m, n, p, s, w, x, y, z, with or without decorations[1], range over N. $*$ denotes a non-member of N and is assumed to satisfy $(\forall n)[n < * < \infty]$. a and b, with or without decorations, range over $N \cup \{*\}$. $\varnothing$ denotes the empty set. $\subseteq$ denotes subset. $\subset$ denotes proper subset. $\supseteq$ denotes superset. $\supset$ denotes proper superset. P and S, with or without decorations, range over sets of N. $\text{card}(S)$ denotes the cardinality of S. $S_1 \varDelta S_2$ denotes the symmetric difference between S_1 and S_2. For $n \in N$ and sets S_1 and S_2, $S_1 =^n S_2$ means that $\text{card}(\{x \mid x \in S_1 \triangle S_2\}) \leqslant n$; $S_1 \Rightarrow^* S_2$ means that $\text{card}(\{x \mid x \in S_1 \varDelta S_2\})$ is finite. D_x denotes the finite set with canonical index x [Rog67].

$\uparrow$ denotes undefined. $\max(\cdot)$, $\min(\cdot)$ denote the maximum and minimum of a set, respectively, where $\max(\varnothing) = 0$ and $\min(\varnothing) = \uparrow$.

η ranges over partial functions with arguments and values from N. $\eta(x)\downarrow$ denotes that $\eta(x)$ is defined; $\eta(x)\uparrow$ denotes that $\eta(x)$ is undefined.

f and g, with or without decorations, range over *total* functions with arguments and values from N. domain (η) and range (η) denote the domain and range of the function η, respectively.

φ denotes a fixed *acceptable* programming system for the partial computable functions: $N \to N$ [Rog58, Rog67, MY78]. φ_i denotes the partial computable function computed by program i in the φ-system. Φ denotes an arbitrary fixed Blum complexity measure [Blu67, HU79] for the φ-system.

W_i denotes domain (φ_i). W_i is, then, the r.e. set/language ($\subseteq N$) accepted (or equivalently, generated) by the φ-program i. $W_i^s \overset{\text{def}}{=} \{x \leqslant s \mid \Phi_i(x) \leqslant s\}$. $\mathscr{E}$ will denote the class of all r.e. sets. L, with or without decorations, ranges over $\mathscr{E}$. $\bar{L}$ denotes the complement of L.

Definition 1. A *hypersimple set* [Pos44] L is an r.e., co-infinite set for which there does not exist a recursive f such that, for all x, $D_{f(x)} \cap \bar{L} \neq \varnothing$, and, for all x, y such that $x \neq y$, $D_{f(x)} \cap D_{f(y)} = \varnothing$.

It is well known that hypersimple sets exist [Rog67]. $\mathscr{L}$, with or without decorations, ranges over subsets of $\mathscr{E}$. The class of finite languages $\mathscr{FIN} \overset{\text{def}}{=} \{L \mid \text{card}(L) < \infty\}$. The class of co-finite sets $\mathscr{COF} \overset{\text{def}}{=} \{L \mid \text{card}(N - L) < \infty\}$.

The quantifiers "$\forall^\infty$", and "$\exists^\infty$" essentially from [Blu67], mean "for all but finitely many" and "there exist infinitely many", respectively.

2.2. *Learning Machines*

We now consider language learning machines. Definition 2 below introduces a notion that facilitates discussion about elements of a language being fed to a learning machine.

[1] Decorations are subscripts, superscripts and the like.

DEFINITION 2. A sequence σ is a mapping from an initial segment of N into $(N \cup \{\#\})$. The *content* of a *sequence* σ, denoted content (σ), is the set of natural numbers in the range of σ. The length of σ, denoted by $|\sigma|$, is the number of elements in σ.

Intuitively, #'s represent pauses in the presentation of data. We let σ and τ, with or without decorations, range over finite sequences. We say that $\sigma \subseteq \tau (\sigma \subset \tau)$ iff σ is an (proper) initial subsequence of τ. We say that $\sigma \supseteq \tau (\sigma \supset \tau)$ iff τ is an (proper) initial subsequence of σ. The sequence $\sigma \cdot \tau$ is obtained by concatenating τ at the end of σ. SEQ denotes the set of all finite sequences. The set of all finite sequences of natural numbers and #'s, SEQ, can be coded onto N.

DEFINITION 3. *A language learning machine* is an algorithmic device which computes a mapping from SEQ into N.

We let $\mathbf{M}$, with or without decorations, range over learning machines.

2.3. *Fundamental Language Identification Paradigms*

DEFINITION 4. A *text* T for a language L is a mapping from N into $(N \cup \{\#\})$ such that L is the set of natural numbers in the range of T. The *content* of a text T. denoted content(T), is the set of natural numbers in the range of T.

Intuitively, a text for a language is an enumeration or sequential presentation of all the objects in the language with the #'s representing pauses in the listing or presentation of such objects. For example, the only text for the empty language is just an infinite sequence of #'s.

We let T, with or without decorations, range over texts. $T[n]$ denotes the finite initial sequence of T with length n. Hence, domain$(T[n]) = \{x \mid x < n\}$. For $n \leqslant |\sigma|$, $\sigma[n]$ denotes the finite initial sequence of σ with length n.

2.3.1. *Finite Vacillatory Language Identification.* Definitions 5 through 8 essentially appear, sometimes for special cases, in [Gol67, OW82a, Cas86, Cas88].

DEFINITION 5. Suppose $\mathbf{M}$ is a learning machine and T is a text. We say $\mathbf{M}(T)$ *converges* (written: $\mathbf{M}(T)\Downarrow) \Leftrightarrow \{M(\tau) \mid \tau \subset T\}$ is finite. If $\mathbf{M}(T)\Downarrow$, then $\mathbf{M}(T)$ is defined $= \{p \mid (\exists^\infty \tau \subset T)[\mathbf{M}(\tau) = p]\}$; otherwise, $\mathbf{M}(T)$ is undefined.

We now introduce criteria for a learning machine to be considered successful on languages.

DEFINITION 6. For $n > 0$, a language learning machine, $\mathbf{M}$, $\mathbf{TxtFex}^a_n$-*identifies* an r.e. language L (written: $L \in \mathbf{TxtFex}^a_n(\mathbf{M})) \Leftrightarrow$ ($\forall$ texts T for L)[$\mathbf{M}(T)\Downarrow$ = a set of cardinality $\leqslant n$ and $(\forall p \in \mathbf{M}(T))[W_p =^a L]$].

DEFINITION 7. A language learning machine, $\mathbf{M}$, $\mathbf{TxtFex}^a_*$-*identifies* an r.e. language L (written $L \in \mathbf{TxtFex}^a_*(\mathbf{M})) \Leftrightarrow$ ($\forall$ texts T for L)[$\mathbf{M}(T)\Downarrow \wedge (\forall p \in \mathbf{M}(T))[W_p =^a L]$].

For $a, b \in N \cup \{*\}$, in $\mathbf{TxtFex}^a_b$-identification, the b is a bound on the number of final grammars and the a a bound on the number of anomalies allowed in these final grammars, *where* a bound of $*$ just means *un*bounded, but *finite*.

DEFINITION 8. $\mathbf{TxtFex}^a_b = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxtFex}^a_b(\mathbf{M})]\}$.

Intuitively, $\mathcal{L} \in \mathbf{TxtFex}^a_b \Leftrightarrow$ there is an effective procedure p such that, if p is given any listing of any language $L \in \mathcal{L}$, it outputs a sequence of grammars *converging* in a non-empty *set* of no more than b grammars, and each of these grammars makes no more than a mistakes in generating L.

$\mathbf{TxtFex}^0_1$-identification is equivalent to Gold's [Gol67] seminal notion of *identification*, also referred to as **TxtEx**-identification in [CL82] and (indirectly) as **INT** in [OW82b, OW82a, OSW86]. $\mathbf{TxtFex}^a_1$-identification is just $\mathbf{TxtEx}^a$-identification from [CL82]. For $n > 0$. $\mathbf{TxtFex}^0_n$-identification is just the notion of $\mathbf{TXTFEX}_n$-identification from [Cas86]. Osherson and Weinstein [OW82a] were the first to define $\mathbf{TxtFex}^0_*$ and $\mathbf{TxtFex}^*_*$. The influence of Gold's paradigm [Gol67] on human language learning is discussed by Pinker [Pin79], Wexler and Culicover [WC80], Wexler [Wex82], and Osherson, Stob, and Weinstein [OSW82, OSW84, OSW86].

2.3.2. *Behaviorally Correct Language Identification.* Next are introduced the cases of success criteria for which the number of final grammars is possibly infinite, not necessarily finite as it is for $\mathbf{TxtFex}^a_b$-identification. Definitions 9 and 10 are from [CL82]. The $a \in \{0, *\}$ cases were independently introduced in [OW82a, OW82b].

DEFINITION 9. A machine $\mathbf{M}$ $\mathbf{TxtBc}^a$-*identifies* L (written: $L \in \mathbf{TxtBc}^a(\mathbf{M})) \Leftrightarrow$ ($\forall$ texts T for L)$(\forall^\infty n)$ $[W_{\mathbf{M}(T[n])} =^a L]$.

DEFINITION 10. $\mathbf{TxtBc}^a = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxtBc}^a(M))]\}$.

We sometimes write **TxtBc** for $\mathbf{TxtBc}^0$.

2.3.3. *Some Basic Results.* We now enumerate the connections between various learning criteria defined above.

THEOREM 1. *For all n, the following hold*;

(a) $\mathbf{TxtFex}^{n+1}_1$-$\mathbf{TxtFex}^n_* \neq \emptyset$.

(b) $\mathbf{TxtFex}^0_{n+2} - \mathbf{TxtFex}^*_{n+1} \neq \emptyset$.

(c) $\mathbf{TxtBc}^{n+1} - \mathbf{TxtBc}^n \neq \emptyset$.

(d) $\mathbf{TxtFex}^{2n+1}_1 - \mathbf{TxtBc}^n \neq \emptyset$.

(e) $\mathbf{TxtFex}^*_1 - \bigcup_n \mathbf{TxtBc}^n \neq \emptyset$.

(f) $\mathbf{TxtBc}\text{-}\mathbf{TxtFex}^*_* \neq \emptyset$.

(g) $\mathbf{TxtBc}^* - (\bigcup_n \mathbf{TxtBc}^n \cup \mathbf{TxtFex}^*_*) \neq \emptyset$.

(h) $\mathbf{TxtFex}^{2n}_* \subset \mathbf{TxtBc}^n$.

(i) $\mathscr{E} \notin \mathbf{TxtBc}^*$.

Most of the above results were announced in [CL82, Cas88] and are all proved in [Cas92]. Recall that $\mathbf{TxtEx}^a \stackrel{\text{def}}{=} \mathbf{TxtFex}^a_1$. Osherson and Weinstein [OW82a] independently proved that $\mathbf{TxtFex}^*_0 \subset \mathbf{TxtFex}^*_*$ and that $\mathbf{TxtFex}^0_1 \subset \mathbf{TxtFex}^0_*$. They also showed that $\mathbf{TxtBc}\text{-}\mathbf{TxtFex}^*_* \neq \emptyset$ and noted that $\mathscr{COF} \in \mathbf{TxtEx}^* - \mathbf{TxtBc}$. We note that the former along with part (c) of the above theorem imply that $\mathbf{TxtEx}^* \subset \mathbf{TxtBc}^*$. [CL82] proved this result by first proving part (c) and then showing that $\mathbf{TxtBc} - \mathbf{TxtEx}^* \neq \emptyset$. They used the self-referential class of languages

$$\mathscr{L}_0 = \{L \mid L \text{ is recursive, infinite and } (\forall^\infty x \in L)[W_x = L]\} \tag{3}$$

to prove the latter result (we further consider this class in Theorem 5). Herein, we achieve the same separation via our characterization of the hypersimple sets in Theorem 3 below.

We now introduce some important definitions and technical concepts which will be used in our proofs.

Let $Progs(\mathbf{M}, \sigma) \stackrel{\text{def}}{=} \{p \mid (\exists n \leqslant |\sigma|)[\mathbf{M}(\sigma[n]) = p]\}$.

Definition 11. We say that σ is a *hypostabilizing sequence for* $\mathbf{M}$ *on* $L \stackrel{\text{def}}{\Leftrightarrow} [[\text{content}(\sigma) \subseteq L] \wedge (\forall \sigma' \supseteq \sigma \mid \text{content}(\sigma') \subseteq L)[\mathbf{M}(\sigma') \in Progs(\mathbf{M}, \sigma)]]$.

Fulk defined the notion of stabilizing sequences [Fu185, Fu190] which differs slightly from the above notion. Next are presented the crucial notions of $\mathbf{TxtEx}^a$, $\mathbf{TxtFex}^a_*$ and $\mathbf{TxtBc}^a$ *locking sequences* which are extensively used in the proofs presented in this paper.

Definition 12. [BB75, OW82a] σ is a $\mathbf{TxtEx}^a$*-locking sequence for* $\mathbf{M}$ *on* $L \stackrel{\text{def}}{\Leftrightarrow} [[\text{content}(\sigma) \subseteq L] \wedge (\forall \sigma' \mid \text{content}(\sigma') \subseteq L \wedge \sigma \subseteq \sigma')[\mathbf{M}(\sigma') = \mathbf{M}(\sigma)] \wedge [\mathbf{W}_{\mathbf{M}(\sigma)} =^a L]]$.

Definition 13. We say that σ is a $\mathbf{TxtFex}^a_*$*-locking sequence for* $\mathbf{M}$ *on* $L \stackrel{\text{def}}{\Leftrightarrow} [[\sigma \text{ is a hypostabilizing sequence for } \mathbf{M} \text{ on } L] \wedge (\forall \sigma' \supseteq \sigma \mid \text{content}(\sigma') \subseteq L)[W_{M(\sigma')} = {}^a L]]$.

Definition 14. We say that σ is a $\mathbf{TxtBc}^a$*-locking sequence for* $\mathbf{M}$ *on* $L \stackrel{\text{def}}{\Leftrightarrow} [[\text{content}(\sigma) \subseteq L] \wedge (\forall \sigma' \supseteq \sigma \mid \text{content}(\sigma') \subseteq L)[W_{\mathbf{M}(\sigma')} = {}^a L]]$.

The following important lemma in learning theory which is essentially due to L. Blum and M. Blum [BB75] will be an important tool used in our proofs in this paper.

Lemma 1 [BB75, OW82a]. *If* $\mathbf{M}$ $\mathbf{TxtEx}^a$*-identifies* L, *then there is a* $\mathbf{TxtEx}^a$*-locking sequence for* $\mathbf{M}$ *on* L.

A similar lemma asserts the existence of $\mathbf{TxtFex}^a_*$ and $\mathbf{TxtBc}^a$-locking sequences. We omit formal statements of these lemmata; nonetheless, we will use these facts in the proofs of Theorem 2 and Theorem 5.

3. RESULTS

Classes of co-finite sets witness many separation results in language learning. For instance, from [CL82, Cas92],

$$\{L \mid L =^{2n+1} N\} \in \mathbf{TxtEx}^{2n+1} - \mathbf{TxtBC}^n. \tag{4}$$

At the same time, other separation results such as $\mathbf{TxtFex}^{n+1}_1\text{-}\mathbf{TxtFex}^n_* \neq \emptyset$ [Cas92] have been proved using self referential classes of languages. We prove the same separation using a natural class of co-finite sets.

Theorem 2. *Let* $\mathscr{L}_{n+1} = \{L \mid L =^{n+1} N\}$. *Then* $\mathscr{L}_{n+1} \in \mathbf{TxtFex}^{n+1}_1 - \mathbf{TxtFex}^n_*$.

Proof. Clearly $\mathscr{L}_{n+1} \in \mathbf{TxtFex}^{n+1}_1$.

Suppose by way of contradiction that $\mathscr{L}_{n+1} \in \mathbf{TxtFex}^n_*(\mathbf{M})$. Then, let σ_N be a $\mathbf{TxtFex}^n_*$ -locking sequence for $\mathbf{M}$ on N. By a variant of Lemma 1, such a σ_N exists. From the definition of locking sequence, it follows that, for all $\sigma \supseteq \sigma_N$, $\mathbf{M}(\sigma) \in Progs(\mathbf{M}, \sigma_N)$. Let $GoodProgs = \{p \in Progs(\mathbf{M}, \sigma_N) \mid W_p =^* N\}$.

We consider 2 cases.

Case 1: $GoodProgs = \emptyset$.

Then clearly $N \notin \mathbf{TxtFex}^n_*(\mathbf{M})$.

Case 2: $GoodProgs \neq \emptyset$.

For $p \in GoodProgs$, let

$$Last(p) = \begin{cases} x_p, & \text{if } [W_p \neq N] \wedge [x_p = \max(N - W_p)]; \\ 0, & \text{otherwise;} \end{cases}$$

Since each $p \in GoodProgs$ satisfies $W_p =^* N$, it is clear that $Last$ is well defined. Let $m = \max(\{Last(p) \mid p \in GoodProgs\})$. Let S be a set of cardinality $n+1$ such that $\min(S) > m$ and $S \cap \text{content}(\sigma_N) = \emptyset$. Let $L = N - S$. Clearly, $L \in \mathscr{L}_{n+1}$. Also, for all $p \in GoodProgs$, it is clear that $L \neq^n W_p$. Thus, for all $p \in Progs(\mathbf{M}, \sigma_N)$, $L \neq^n W_p$. Let $T \supset \sigma_N$ be any text for L. It is clear that $\mathbf{M}$ does not $\mathbf{TxtFex}^n_*$-identify L on T.

Thus, $\mathscr{L}_{n+1} \notin \mathbf{TxtFex}^n_*$. ∎

$\mathscr{COF}$ does not witness that $\mathbf{TxtBc}^*$ separates from $\mathbf{TxtEx}^*$, i.e., $\mathscr{COF} \notin \mathbf{TxtBc}^* - \mathbf{TxtEx}^*$. Perhaps if $\mathscr{COF}$ were augmented by another language the result would witness the separation. This partly motivates the following. Intuitively we think of $\mathscr{I}$ in Definition 15 as an identification criterion.

Definition 15. Let $\mathscr{I}$ be a set of classes of languages.

(a) We say L is $\mathscr{I}$*-admissible* to $\mathscr{L} \stackrel{\text{def}}{\Leftrightarrow} \mathscr{L} \cup \{L\} \in \mathscr{I}$. We say L is $\mathscr{I}$*-inadmissible* to $\mathscr{L} \stackrel{\text{def}}{\Leftrightarrow} L$ is not $\mathscr{I}$*-admissible* to $\mathscr{L}$.

(b) We say that $\mathscr{L}$ is $\mathscr{I}$-*saturated* $\stackrel{\text{def}}{\Leftrightarrow}$ $[[\mathscr{L} \in \mathscr{I} \wedge (\exists L)$ $[L$ is $\mathscr{I}$-inadmissible to $\mathscr{L}]]$.

(c) We say that $\mathscr{L}$ is $\mathscr{I}$-*supersaturated* $\stackrel{\text{def}}{\Leftrightarrow}$ $[[\mathscr{L} \in \mathscr{I}] \wedge$ $(\forall L \in \mathscr{E} - L)[L$ is $\mathscr{I}$-inadmissible to $\mathscr{L}]]$.

[OSW86] define variants of the saturatedness and supersaturatedness notions above (they call their variants 'maximal' and 'saturated' respectively). Their definitions pertain to not necessarily algorithmic language learning.

As an example of supersaturated language classes, consider $\mathscr{FIN}$, the class of all finite languages. Clearly, $\mathscr{FIN} \in$ **TxtEx***. Also, for *any* infinite language L, it can be proved that $\mathscr{FIN} \cup \{L\} \notin$ **TxtEx*** [OW82a, CL82]. Thus, $\mathscr{FIN}$ is **TxtEx***-supersaturated (it can also be proved that it is **TxtBc***-supersaturated). In fact, it is easily proved that $\mathscr{FIN}$ is the only **TxtEx***-supersaturated class of languages [OSW86]. On the other hand, from our next and surprising main theorem, one can see that (all and) only the hypersimple sets are **TxtEx***-inadmissible to $\mathscr{COF}$.

THEOREM 3. *L hypersimple $\Leftrightarrow$ L is* **TxtEx***-*inadmissible to* $\mathscr{COF}$.[2]

Proof. ($\Rightarrow$) Suppose L is hypersimple. Suppose by way of contradiction that $\mathscr{COF} \cup \{L\} \in \mathbf{TxtEx}^*(\mathbf{M})$. Then, let σ_L be a locking sequence for $\mathbf{M}$ on L. By Lemma 1, such a σ_L exists. Let g be a recursive function such that, for all σ, $D_{g(\sigma)} = \text{content}(\sigma)$. Now consider program p which on all inputs x computes as specified below. $\varphi_p(0) = g(\min(\bar{L}))$.

For $x \geqslant 1$, $\varphi_p(x)$ is computed as follows.

1. First compute $\varphi_p(x-1)$.

2. Search for σ such that $\text{content}(\sigma) \subset \{y \mid y \geqslant 1 + \max(D_{\varphi_p(x-1)})\}$ and $\mathbf{M}(\sigma_L \cdot \sigma) \neq \mathbf{M}(\sigma_L)$. If (at all) such a σ is found, let $\varphi_p(x) = g(\sigma)$.

This completes the specification of program p. We now consider 2 cases.

Case 1: φ_p is not total.

Let x be the least value such that $\varphi_p(x)\uparrow$. Clearly $x \geqslant 1$. Since x is the least value such that $\varphi_p(x)\uparrow$, step 1 in the computation of $\varphi_p(x)$ terminates. Let $m = 1 + \max(D_{\varphi_p(x-1)})$. Let T' be a text for the language $\{x \mid x \geqslant m\}$. Let $T = \sigma_L . T'$. T is clearly a text for the co-finite set $L' = \text{content}(\sigma_L) \cup \{x \mid x \geqslant m\}$. Now $\mathbf{M}(T) = \mathbf{M}(\sigma_L)$ (otherwise, step 2 in the computation of program p on input x would have terminated as well; hence, $\varphi_p(x)$ would have been defined). Thus, from our supposition that $\mathscr{COF} \cup \{L\} \in \mathbf{TxtEx}^*(\mathbf{M})$, it follows that $L =^* W_{\mathbf{M}(\sigma_L)} = W_{\mathbf{M}(T)} =^* L'$. Thus we conclude that $L' =^* L$, which is a contradiction since L' is co-finite and L, a hypersimple set, is co-infinite.

Case 2: φ_p is total.

Since σ_L is a locking sequence for $\mathbf{M}$ on L, it follows from the specification of program p that for all x, $D_{\varphi_p(x)} \cap \bar{L} \neq \emptyset$ and that for all x, y such that $x \neq y$, $D_{\varphi_p(x)} \cap D_{\varphi_p(y)} = \emptyset$. Thus, $f = \varphi_p$ witnesses that L is not hypersimple which is a contradiction.

($\Leftarrow$) Suppose that L is not hypersimple. If L is co-finite, then clearly $\mathscr{COF} \cup \{L\} = \mathscr{COF} \in$ **TxtEx***. So, suppose L is not co-finite. Then there exists a recursive f such that, for all x, $D_{f(x)} \cap \bar{L} \neq \emptyset$ and for x, y such that $x \neq y$, $D_{f(x)} \cap D_{f(y)} = \emptyset$. Let p_L and p_N be such that $W_{p_L} = L$ and $W_{p_N} = N$. Consider $\mathbf{M}$ defined as follows.

$$\mathbf{M}(\sigma) = \begin{cases} p_N, & \text{if } (\exists x \leqslant |\sigma|)[D_{f(x)} \subseteq content(\sigma)]; \\ p_L, & \text{otherwise.} \end{cases}$$

Let T be a text for a language in $\mathscr{COF} \cup \{L\}$. We consider 2 cases.

Case 1: T is a text for L.

Since it is the case that for all x, $D_{f(x)} \cap \bar{L} \neq \emptyset$, it follows that $(\forall \sigma \subset T)(\forall x)[D_{f(x)} \nsubseteq \text{content}(\sigma)]$. Thus, for all n, $\mathbf{M}(T[n]) = p_L$. Thus, $\mathbf{M}$ **TxtEx***-identifies L.

Case 2: T is a text for $L' \in \mathscr{COF}$.

Since L' is co-finite, it follows that $(\forall^\infty x \in \bar{L})[x \in L']$ and $(\forall^\infty x \in L)[x \in L']$. Note also that $\text{card}(x \mid (x \in \bar{L}$ and $(\exists y)[x \in D_{f(y)}])$ is infinite. Thus, it is clear that there exists n such that $(\exists x \leqslant n)[D_{f(x)} \subseteq \text{content}(T[n])]$. From the definition of $\mathbf{M}$, it follows that for all $n' \geqslant n$, $\mathbf{M}(T[n']) = p_N$. Since L' was an arbitrary co-finite language, it is clear that $\mathbf{M}$ **TxtEx***-identifies $\mathscr{COF}$.

Thus $\mathbf{M}$ **TxtEx***-identifies $\mathscr{COF} \cup \{L\}$.

COROLLARY 1. $\mathscr{COF}$ *is* **TxtEx***-*saturated, but not* **TxtEx***-*supersaturated.*

Proof. By Theorem 3, any hypersimple set is **TxtEx***-inadmissible to $\mathscr{COF}$, yet r.e., nonhypersimple, co-infinite sets are **TxtEx***-admissible. ∎

The following corollary can be proved by a straightforward modification of the proof of Theorem 3.

COROLLARY 2. $\mathscr{COF}$ *is* $\mathbf{TxtFex}^*_*$-*saturated, but not* $\mathbf{TxtFex}^*_*$-*supersaturated.* ∎

THEOREM 4. $\mathscr{COF}$ *is not* **TxtBc***-*saturated.*

Proof. Let L be any arbitrary language. We will prove that $\mathscr{COF} \cup \{\mathscr{L}\} \in$ **TxtBc***.

Let p_L be such that $W_{p_L} = L$. Then, let f be a recursive function such that, for all σ, $W_{f(\sigma)}$ is defined as follows.

[2] Let $\mathbf{ExGen}^a$ [CL82] be the learning criterion just like $\mathbf{TxtEx}^a$ except that the inputs are characteristic functions of (rather than texts for) languages. Frank Stephan, after he saw this theorem, basing himself also on Lemma 11.3 of [KS93], pointed out to us that he can prove: L is hypersimple $\Leftrightarrow$ $(\mathscr{COF} \cup \{L\}) \notin \mathbf{ExGen}^0 \Leftrightarrow (\mathscr{COF} \cup \{L\}) \notin \mathbf{ExGen}^*$.

1 initially $W_{f(\sigma)}$ is empty.

2. $i=0$.
   repeat
   Enumerate i into $W_{f(\sigma)}$.
   $i=i+1$.
   until content$(\sigma) \subseteq W^i_{p_L}$

3. Enumerate all the elements of W_{p_L} into $W_{f(\sigma)}$.

Suppose σ is such that content$(\sigma) \subseteq L$. Then, there exists an i such that content $(\sigma) \subseteq W^i_{p_L}$. Thus, in the enumeration of $W_{f(\sigma)}$, the **repeat**-loop in step 2 terminates giving $W_{f(\sigma)} =^* L$.

Now suppose σ is such that content $(\sigma) \nsubseteq L$. Then, in the enumeration of $W_{f(\sigma)}$, the **repeat**-loop in step 2 does not terminate. Thus, $W_{f(\sigma)} = N$.

Consider **M** such that, for all σ, $\mathbf{M}(\sigma) = f(\sigma)$. From the preceding assertions, it is clear that **M** **TxtBc***-identifies $\mathscr{COF} \cup \{L\}$. ∎

In fact, the proof of Theorem 4 can be generalized to prove the following. We omit details here.

Corollary 3. $(\forall$ *finite* $\mathscr{L})[\mathscr{COF} \cup \mathscr{L} \in \mathbf{TxtBc}^*]$.

The following separation result from [CL82, OW82a] follows as a corollary of Theorems 3 and 4.

Corollary 4. $\mathbf{TxtBc}^* - \mathbf{TxtEx}^* \neq \emptyset$.

Proof. Let L be a hypersimple set. Let $\mathscr{L} = \mathscr{COF} \cup \{L\}$. From Theorem 4, $\mathscr{L} \in \mathbf{TxtBc}^*$ and from Theorem 3, $\mathscr{L} \notin \mathbf{TxtEx}^*$. ∎

$\mathscr{L}_0$ is defined in (3) in Section 2.3.3 above. $\mathscr{L}_0$ is the self referential class used in [CL82] to witness Corollary 4 just above.

Theorem 5. *$\mathscr{L}_0$ is* **TxtBc***-*saturated.*

Proof. We observe first that $\mathscr{COF} \cap \mathscr{L}_0 = \emptyset$ (since any co-finite set contains infinitely many indices for $L_1 = \emptyset$ and $L_2 = N \neq^* L_1$). Now we prove the following claim.

Claim 1. $(\forall$ *finite* $S)(\exists L \in \mathscr{L}_0)[S \subset L]$.

Proof. Fix S. Then, by the operator recursion theorem [Cas74], there exists a monotone increasing recursive function f such that, for all x: $W_{f(x)} = S \cup \{f(w) \mid w \in N\}$.

Let $L = S \cup \{f(w) \mid w \in N\}$. Since $L =^* \text{range}(f)$, for monotone increasing recursive function f, it is clear that L is infinite and recursive. Also, $(\forall^\infty x \in L)[W_x = L]$. Hence $L \in \mathscr{L}_0$. This completes the proof of Claim 1. ∎

We now prove that N is **TxtBc***-inadmissible to $\mathscr{L}_0$. Suppose by way of contradiction that $\mathscr{L}_0 \cup \{N\} \in \mathbf{TxtBc}^*(\mathbf{M})$. Then, let σ_N be a **TxtBc***-locking sequence for **M** on N.

Let $L \in \mathscr{L}_0$ be such that contents $(\sigma_N) \subset L$. From Claim 1, such an L exists. Let $T \supset \sigma_N$ be a text for L. Since $L \notin \mathscr{COF}$, it follows that $(\forall^\infty n)[W_{\mathbf{M}(T[n])} =^* N \neq^* L]$. ∎

Corollary 5. *For all n, $\mathscr{L}_0$ is* $\mathbf{TxtBc}^n$-*saturated.*

Proof. This corollary follows from Theorem 5 and the fact that, for all n, $\mathscr{L}_0 \in \mathbf{TxtBc}^n \subset \mathbf{TxtBc}^*$. ∎

Next we explore analogs of Theorems 3 and 4, namely Theorems 6 and 7, respectively, for **TxtEx** in place of **TxtEx*** and **TxtBc** in place of **TxtBc***.

Definition 16. Let $\text{FinMeet}(L) \stackrel{\text{def}}{=} \{S \text{ finite} \mid S \cap L \neq \emptyset\}$.

Theorem 6. *For all $L \in \mathscr{E}$, L is recursive $\Leftrightarrow L$ is* **TxtEx**-*admissible to* $\text{FinMeet}(\bar{L})$.

Proof. $(\Rightarrow)$ Fix $L \in \mathscr{E}$. Now suppose L is recursive. We will construct machine **M** which **TxtEx**-identifies $\text{FinMeet}(\bar{L}) \cup \{L\}$. Let g be a recursive function such that, for all S, $D_{g(S)} = S$. Let i_0 be such that $W_{i_0} = L$.

For all σ, let

$$\mathbf{M}(\sigma) = \begin{cases} g(\text{content}(\sigma)) & \text{if content}(\sigma) \cap \bar{L} \neq \emptyset; \\ i_0, & \text{otherwise;} \end{cases}$$

Note that the the predicate "content$(\sigma) \cap \bar{L} \neq \emptyset$" is recursively testable since L is recursive. It is clear that **M** **TxtEx**-identifies $\text{FinMeet}(\bar{L}) \cup \{L\}$.

$(\Leftarrow)$ Suppose L is r.e. but not recursive. Suppose by way of contradiction that L is **TxtEx**-admissible to $\text{FinMeet}(\bar{L})$. Then there exists **M** such that **M** **TxtEx**-identifies $\text{FinMeet}(\bar{L}) \cup \{L\}$. Let σ be a **TxtEx**-locking sequence for **M** on L. For all i, let T_i be the text which starts with σ followed by an infinite sequence of i. Since **M** **TxtEx**-identifies $\text{FinMeet}(\bar{L}) \cup \{L\}$, it can be deduced that for all i, $i \in \bar{L} \Leftrightarrow (\exists > |\sigma|)[\mathbf{M}(T_i[j]) \neq \mathbf{M}(\sigma)]$. Thus $\bar{L}$ is r.e., contradicting our supposition.

Theorem 7. *For all L,* $\text{FinMeet}(\bar{L}) \cup \{L\} \in \mathbf{TxtBc}$.

Proof. Let L be any arbitrary language and let p_L be such that $W_{p_L} = L$. Then, let f be a recursive function such that, for all σ, $W_{f(\sigma)}$ is defined as follows.

1. Enumerate into $W_{f(\sigma)}$ all elements in content (σ).

2. $i=0$.
   repeat
   $i=i+1$.
   until content$(\sigma) \subseteq W^i_{p_L}$

3. Enumerate all the elements of W_{p_L} into $W_{f(\sigma)}$.

Suppose σ is such that content$(\sigma) \subseteq L$. Then, there exists an i such that content$(\sigma) \subseteq W^i_{p_L}$. Thus, the **repeat**-loop in step 2 terminates giving $W_{f(\sigma)} = \text{content}(\sigma) \cup L = L$.

Now suppose σ is such that content$(\sigma) \nsubseteq L$. Then, in the enumeration of $W_{f(\sigma)}$, the **repeat**-loop in step 2 does not terminate. Thus, $W_{f(\sigma)} = \text{content}(\sigma)$.

Consider **M** such that, for all σ, $\mathbf{M}(\sigma)=f(\sigma)$. From the preceding assertions, it is clear that **M** **TxtBc**-identifies $\mathrm{FinMeet}(\bar{L})\cup\{L\}$. ▮

Lastly we derive the following fact from [OW82a, CL82], employing Theorems 6 and 7.

COROLLARY 6. $\mathbf{TxtBc}-\mathbf{TxtEx}\neq\varnothing$

Proof. Fix L, a r.e. not recursive language and let $\mathscr{L}=\mathrm{FinMeet}(\bar{L})\cup\{L\}$. From Theorems 6 and 7, it is clear that $\mathscr{L}\in\mathbf{TxtBc}-\mathbf{TxtEx}$. ▮

REFERENCES

[BB75] L. Blum and M. Blum, Toward a mathematical theory of inductive inference, *Inform. and Control* **28** (1975), 125–155.

[BC93] G. Baliga and J. Case, Learnability: Admissible, cofinite and hypersimple sets, *in* "Proceedings of the 20th International Colloquium on Automata, Languages and Programming," Vol. 700, pp. 289–300, Springer-Verlag, Sweden, 1993.

[Blu67] M. Blum, A machine independent theory of the complexity of recursive function, *J. ACM* **14** (1967), 322–336.

[Cas74] J. Case, Periodicity in generations of automata, *Math. Systems Theory* **8** (1974), 8–15.

[Cas86] J. Case, Learning machines, *in* "Language Learning and Concept Acquisition" (W. Demopoulos and A. Marras, Eds.), Ablex 1986.

[Cas88] J. Case, The power of Vacillation, *in* "Proceedings of the Workship on Computational Learning Theory" (D. Haussler and L. Pitt, Eds.), pp. 133–142, Morgan Kaufmann, 1988; journal Version being revised for possible publication.

[Cas92] J. Case, "The Power of vacillation in Language Learning," Tech. Rep. 93-08, University of Delaware, 1992, expands on [Cas88]; journal article under revision for possible publication.

[CL82] J. Case and C. Lynes, Machine inductive inference and language identification, *in* "Proceedings of the 9th International Colloquium on Automata, Languages and Programming" (M. Nielsen and E. Schmidt, Eds.), Vol. 140, pp. 107–115, Springer-Verlag, Berlin, 1982.

[Fu185] M. Fulk, "A Study of Inductive Inference Machines," Ph.D. thesis, SUNY at Buffalo, 1985.

[Ful90] M. Fulk, Prudence and other conditions on formal language learning, *Inform. and Comput.* **85** (1990), 1–11.

[Gol67] E. Gold, Language identification in the limit, *Inform. and Control* **10** (1967), 447–474.

[HU79] J. Hopcroft and J. Ullman, "Introduction to Automata Theory Languages and Computation," Addison–Wesley, 1979.

[JY81] D. Joseph and P. Young, Independence results in computer science, *J. Comput. System Sci.*, (1981), 205–222.

[Kre57] G. Kreisel, Independent recursive axiomatizability, *J. Symbolic Logic* **22** (1957), 22–109.

[KS93] M. Kummer and F. Stephan, On the structure of degrees of inferability, *in* "Proceedings of the Sixth Annual Conference on Computational Learning Theory, Santa Cruz, California," pp. 117–126, ACM Press, July 1993; *J. Comput. System Sci.*, (Special Issue COLT'93), to appear.

[McN66] D. McNeill, Developmental psycholinguistics, *in* "The Genesis of Language" (F. Smith and G. A. Miller, Eds.), pp. 15–84, MIT Press, Cambridge, 1966.

[MY78] M. Machtey and P. Young, "An Introduction to the General Theory of Algorithms," North Holland, New York, 1978.

[OSW82] D. Osherson, M. Stob, and S. Weinstein, Ideal learning machines, *Cognitive Sci.* **6** (1982), 277–290.

[OSW84] D. Osherson, M. Stob, and S. Weinstein, Learning theory and natural language, *Cognition* **17** (1984), 1–28.

[OSW86] D. Osherson, M. Stob, and S. Weinstein, "Systems that Learn, An Introduction to Learning Theory for Cognitive and Computer Scientists," MIT Press, Cambridge, MA, 1986.

[OW82a] D. Osherson and S. Weinstein, Criteria of language learning, *Informat. Control* **52** (1982), 123–138.

[OW82b] D. Osherson and S. Weinstein, A note on formal learning theory, *Cognition* **11** (1982), 77–88.

[PE68] M. Pour-El, Independent axiomatization and its relation to the hypersimple set, *Z. Math. Logik Grundlagen Math.* **14** (1968), 449–456.

[Pin79] S. Pinker, Formal models of language learning, *Cognition* **7** (1979), 217–283.

[Pos44] E. Post, Recursively enumerable sets of positive integers and their decision problems, *Bull. Amer. Math. Soc.* **50** (1944) 284–316.

[Rog58] H. Rogers, Gödel numberings of partial recursive functions, *J. Symbolic Logic* **23** (1958) 331–341.

[Rog67] H. Rogers, "Theory of Recursive Functions and Effective Computability," McGraw–Hill, New York, 1967; reprinted, MIT Press, Cambridge, MA, 1987.

[Roy87] J. Royer, "A Connotational Theory of Program Structure," Lecture Notes in Computer Science, Vol. 273, Springer-Verlag, New York/Berlin, 1987.

[WC80] K. Wexler and P. Culicover, "Formal Principles of Language Acquisition," MIT Press, Cambridge, MA, 1980.

[Wex82] K. Wexler, On extensional learnability, *Cognition* **11** (1982) 89–95.